



567 - Risk

Link:

https://uva.onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&page=show_problem&problem=508

Solution:

C++	http://ideone.com/NjJK6U
Java	https://ideone.com/ualmqa
Python	https://ideone.com/N401ba

Tóm tắt đề: Đề bài cho 20 thành phố, tìm đường đi ngắn nhất của một số cặp thành phố, trên đồ thị vô hướng có trọng số mỗi đường đi là 1. Có nhiều bộ test nhưng không cho trước số bộ test là bao nhiêu người dùng phải đọc hết toàn bộ dữ liệu mới biết có bao nhiêu bộ test.

Input

Có nhiều bộ test, mỗi bộ test gồm có:

- 19 dòng: mỗi dòng đại diện cho mỗi đỉnh. (ví dụ dòng đầu tiên là đỉnh 1, dòng 2 là đỉnh 2...) mỗi dòng số đầu tiên là số lượng đỉnh (count) có kết nối với đỉnh mà nó đang xét. "count" số tiếp theo cách nhau bởi dấu khoảng trắng lần lượt là các đỉnh đang có kết nối với đỉnh index đang xét.
- Số tiếp theo 19 dòng trên là số N là số lượng cặp đỉnh đề bài yêu cầu tìm đường đi ngắn nhất.
- N dòng tiếp theo lần lượt là các cặp cần tìm đường đi ngắn nhất.

Output

Mỗi bộ test in ra đúng định dạng như sau:

Cẩn thận in ra đúng định dạng, nếu số nào có 1 chữ số thì phải in ra thêm khoảng trắng để cho mỗi hàng cân đối với số có 2 chữ số. Kết thúc mỗi testcase in ra dấu xuống dòng. Kết thúc cả bộ test là 2 dấu xuống dòng.

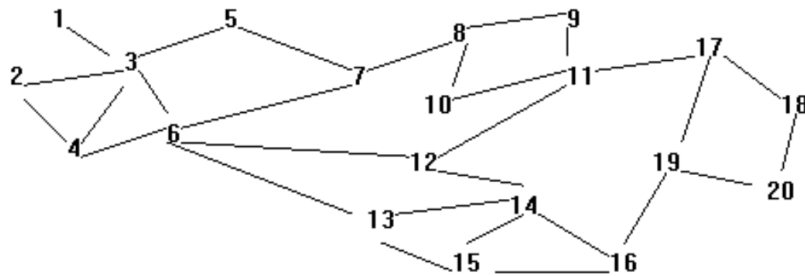
Hướng dẫn làm bài: Dùng Floyd-Warshall để tìm đường đi ngắn nhất từ một đỉnh đến tất cả các đỉnh là ra kết quả, bài này không khó nhưng quan trọng là cách đọc INPUT vào và xuất kết quả cho đúng định dạng

1 3	Test Set #1
2 3 4	1 to 20: 7
3 4 5 6	2 to 9: 5
1 6	19 to 5: 6
1 7	18 to 19: 2
2 12 13	16 to 20: 2
1 8	
2 9 10	Test Set #2
1 11	1 to 20: 4
1 11	8 to 20: 5
2 12 17	15 to 16: 2
1 14	11 to 4: 1
2 14 15	7 to 13: 3
2 15 16	2 to 16: 4
1 16	
1 19	
2 18 19	
1 20	
1 20	
5	
1 20	
2 9	
19 5	
18 19	
16 20	
4 2 3 5 6	
1 4	
3 4 10 5	
5 10 11 12 19 18	
2 6 7	
2 7 8	
2 9 10	
1 9	
1 10	
2 11 14	
3 12 13 14	
3 18 17 13	
4 14 15 16 17	
000	
2 18 20	
1 19	
1 20	
6	
1 20	

8 20 15 16 11 4 7 13 2 16	
---------------------------------------	--

Giải thích: Giải thích bộ test đầu tiên:

19 dòng đầu tiên lần lượt là thông tin của 19 đỉnh có kết nối với nhau. Trong testcase đầu tiên đồ thị hiển thị như sau:



Tìm đường đi ngắn nhất từ:

1 20, 2 9, 19 5, 18 19, 16 20.

Hướng dẫn giải: Từ danh sách cạnh, tạo đồ thị, sau đó chạy Dijkstra từ đỉnh U. Sau khi chạy xong dựa vào mảng chi phí dist để tìm chi phí đến các đỉnh cần tìm. Nếu đỉnh nào chi phí là vô cực (INF) thì sẽ không có đường đi đến đó.

Độ phức tạp: $O(T * N^3)$ với $N = 20$ là số lượng thành phố, và T là số lượng bộ dữ liệu đầu vào.